

SLIDEFINDER: Herramienta docente para clasificar, buscar y compartir imágenes en soporte electrónico

José-María Zamanillo, Daniel Franco, Isabel Zamanillo, Constantino Pérez-Vega.

jose.zamanillo@unican.es, uc331@alumnos.unican.es, uc8915@alumnos.unican.es, perevr@unican.es

Dpto. de Ingeniería de Comunicaciones Universidad de Cantabria

ETSI y Telecomunicación Av. de los Castros s/n.39005 Santander (Cantabria).

Abstract- This paper outlines the joint project developed by the University of Cantabria and the Non Government Association “Educación para la Solidaridad Grupo Brezo” in order to develop a software tool to classify, search and find images thru the World Wide Web. The main idea of the project implies the digitalization, storage and sharing of more than 80,000 slides of different themes (landscapes, people, folklore, history, natural disasters, etc.) made along 40 years by the members of Brezo all over the world. This kind of images can be of interest for educational, E-learning and research purposes. An open source platform based in PHP and MySQL has been chosen in order to develop the application shown here, the engine has been implemented under “friendly user” philosophy and it is self-explicative, the above mentioned features make the software an interesting tool easy to use by users at any level.

I. INTRODUCCIÓN

En 2005 la Universidad de Cantabria a través del Departamento de Ingeniería de Comunicaciones y la ONG Grupo Brezo firmaron un convenio cofinanciado por la Consejería de relaciones Institucionales y Asuntos Europeos del Gobierno de Cantabria a través de la Dirección General de Asuntos Europeos y Cooperación al Desarrollo para crear un archivo electrónico digital capaz de almacenar un fondo que se encontraba en soporte analógico de 80.000 diapositivas de 35 mm propiedad de la ONG Grupo Brezo. Las diapositivas, han de ser previamente digitalizadas y post-procesadas bajo soporte informático antes de ser incorporadas al archivo definitivo, el cual en su versión de Internet será accesible de forma gratuita para todos los centros de enseñanza de todo el mundo así como por aquellas organizaciones sin ánimo de lucro que lo deseen.

La idea consiste no solo en la digitalización propiamente dicha del archivo en soporte analógico, si no en el desarrollo de una herramienta informática que acceda al archivo de fácil acceso por parte del usuario dicha herramienta ha sido bautizada con el nombre de SlideFinder. De esta manera cuando el archivo este finalizado este pueda ser consultado bajo diferentes soportes: INTERNET, CD-ROM, DVD-ROM y plataformas basadas en diferentes sistemas operativos: Windows, Macintosh, Unix, etc. La temática y variedad de las mismas es muy diversa abarcando todo tipo de temas de índole regional, nacional e internacional.

Cuando se planteó la posibilidad de realizar este trabajo la idea de partida era realizar una base de datos mixta de imágenes mas texto (la base debía incluir un pequeño texto explicativo de hasta 256 caracteres el cual actúa como pie de foto). Aunque el planteamiento parezca sencillo, realmente

no ha sido así, ya que han aparecido incompatibilidades de todo tipo, no sólo entre el software y las plataformas utilizadas sino en la manera de implementar la base de datos.

II. PLANTEAMIENTO DEL PROBLEMA

Generalmente, cuando se desarrolla una aplicación informática para la red, se programa la misma conociendo de antemano donde va a alojarse dicha página web, sobre qué sistema operativo va a ejecutarse y como va a estar definido el sistema de archivos. Es decir, la página se realiza para que sea única, mientras que en el caso que nos ocupa, se requería que todo el sistema pudiera trasladarse de un ordenador a otro con los mínimos problemas posibles de instalación y funcionamiento. Entre las distintas posibilidades estudiadas estuvo la idea de realizar un CD de arranque con sistema operativo embebido que arrancara automáticamente todo el sistema. Descartado en este caso realizarlo sobre una plataforma Windows por su tiempo de arranque, se barajó la posibilidad de utilizar un sistema operativo Linux embebido. Esta idea tenía una gran ventaja, cuando siempre se utilice el mismo sistema operativo, con la misma configuración, con lo que no habría que realizar ningún cambio en el ordenador donde se quisiera utilizar el programa. Pero esta idea quedo finalmente descartada porque impedía la compatibilidad entre el sistema operativo del usuario y el Linux embebido, es decir, el usuario cuando estuviera trabajando la aplicación sólo podría realizar las funciones implementadas sin que él pudiera utilizar ninguno de sus programas instalados, ni reutilizar los datos y consultas que hubiese obtenido de la aplicación.

Finalmente, una vez decidido que el sistema debería ser totalmente accesible vía web, fue necesario decidir dos elementos básicos para la aplicación: el lenguaje de programación y la base de datos a utilizar. Para la base de datos, se estudiaron distintas posibilidades decidiéndose finalmente por la base de datos MySQL [1-4], la cual se basa en el lenguaje SQL, de libre distribución. Nuestro grupo es consciente de que, quizás no sea la mejor y mas completa base de datos del mercado pero si la que mejor se adecuaba a las necesidades del proyecto. Probablemente, una de las mejores opciones multiplataforma puede ser la base de datos de Oracle®, cuya principal desventaja es su alto coste por licencia y por ello suele ser utilizada por entidades financieras y en grandes bases de datos de empresas de todo tipo. Otra posibilidad consiste en utilizar la base de datos SQL Server, la cual se basa en el lenguaje SQL [5] igual que

MySQL. Aunque SQL es más económica que Oracle, presenta la desventaja que al estar desarrollada por Microsoft sólo funciona en entorno Windows con lo cual quedaba descartada para el desarrollo de la presente aplicación. Existen en el mercado otras bases de datos de características semejantes a las comentadas como: Progress, IBM-DB2 o Informix. Pero no cabe duda que MySQL es la que la que mejor relación calidad-precio ofrece, y por tanto se adapta perfectamente a los objetivos del presente trabajo.

III. EL PROGRAMA SLIDEFINDER

En el siguiente epígrafe se comentaran los aspectos relativos a la programación, organización y funcionamiento interno de la aplicación *SlideFinder*.

A. Programación de la base de datos

En la programación de la aplicación se han utilizado dos bases de datos diferenciadas: una interna la cual utiliza el motor MySQL y la otra externa la cual se ha denominado con el mismo nombre que la propia aplicación *SlideFinder*. La razón es obvia, y va encaminada a aumentar la seguridad de la aplicación, e impedir que algún usuario malintencionado pueda acceder a la base de datos del motor MySQL y obtenga las claves de administrador y así tener acceso a modificar los datos del ordenador central, que será el único computador desde el que se puedan agregar, borrar o alterar elementos y registros en la base de datos Slide-Finder.

La primera idea estudiada fue realizar una única tabla con todos los datos necesarios, esto simplificaría en extremo la labor del programador, ya que las consultas serían realmente breves, pero por desgracia disparaba casi exponencialmente el tiempo de consulta y aumentaba en exceso el código de programación, siendo por ello descartada. Posteriormente, se decidió incluir las propias imágenes en la base de datos, lo cual facilita el trabajo de consulta. Pero ello conlleva dos aspectos negativos:

- Disminución de la velocidad de acceso a los datos debido al aumento de tamaño de los registros.
- Imposibilidad transportar la aplicación cuando la base de datos, sea de tamaño grande debido al elevado número de datos almacenados.

Además se ha decidido incluir en la base de datos el nombre del archivo así como su localización en el disco duro y en cuanto a las tablas, cuanto más se dividan menos redundancia de datos existe, consiguiéndose de esta forma reducir el campo de búsqueda con el correspondiente incremento de velocidad de ejecución. La Fig. 1 muestra el diagrama de flujo, y la estructura de tablas de la base de datos, además de sus interconexiones. Los colores (o tonos de grises si se visualiza la Fig.1 en blanco y negro) indican la interdependencia entre las diferentes tablas, así como la relación entre ambas. Puede observarse que existe una tabla aislada que no tiene relación con la estructura general, ello es debido a que actuará solo como apoyo de las demás. Todas las tablas son del tipo MyISAM con lo que las relaciones creadas entre las tablas no son vinculantes, es decir, si se borrara un país (España) y las tablas fueran vinculantes, todas sus provincias, ciudades, así como todos los datos de las imágenes se borrarían automáticamente de la base de datos. Esto es muy peligroso y por lo tanto se decidió no utilizar tablas del tipo relacionales (InnoDB). Aunque las tablas del tipo MyISAM obligan a propagar los cambios, ello siempre

es mejor que arriesgarse a tener que volver a introducir todos los datos de nuevo.

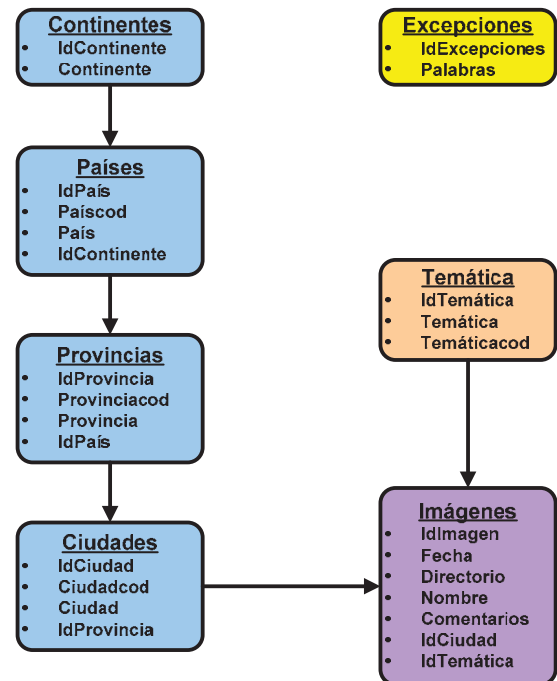


Fig. 1. Estructura de la base de datos principal.

Además, como se muestra en la Fig. 1, las tablas tienen un campo principal con un identificador por registro, este campo es único y actúa como identificador de cada elemento. En casi todas las tablas (menos en imágenes y excepciones) existe un identificador con la estructura “nombre tabla” + “cod”, que genera el nombre del fichero y además actualiza automáticamente la base de datos.

B. Imágenes

Las imágenes pueden almacenarse en cualquier medio mientras esté conectado físicamente al ordenador, ya sea un disco duro interno o extraíble, CD, DVD, pendrive, etc. Esto es importante en aras de la portabilidad del sistema. Se ha fijado como tamaño estándar para la presentación de las imágenes en la Web el de 640x480 pixels, color real¹, y formato JPG pero eso no quiere decir que las imágenes deban tener ese tamaño donde estén almacenadas. A la hora consultar el tamaño no va a afectar al proceso. Donde si se notará el efecto será en los clientes cuando descarguen la imagen a través de su conexión a Internet. En la Fig. 2, se muestra la forma de identificar los ficheros que contienen las imágenes y el tipo de tabla pertenece dicha información en la base de datos. Todos los nombres de fichero de imágenes deben respetar esta estructura y tener el mismo número de caracteres. En el caso, de desconocer alguna información bastará con identificar esa parte con el carácter “x”. El nombre del archivo debe ser único no permitiéndose la posibilidad de agregar dos imágenes con el mismo nombre aún en directorios diferentes. La organización de los ficheros que contienen las imágenes en el disco duro, es bastante flexible, debiendo estar organizados en directorios, aunque sólo con un nivel de profundidad, esto quiere decir, que no se

¹ Se entiende por color real una profundidad de color de 2³² bits.

permiten directorios de imágenes anidados, teniendo que estar todos estos directorios dentro de un directorio principal llamado “imágenes”. La estructura es la siguiente:

{ruta previa}/imagenes/{directorio}/{nombre imagen}.jpg

El número de carpetas así como el número de imágenes dentro de cada directorio, queda limitado por los recursos del sistema anfitrión del E-archivo. La base de datos desarrollada almacena las fechas en el formato anglosajón, por lo que, la fecha ha de ser extraída del nombre de la imagen y transformada del formato hispano al anglosajón antes de ser introducida en la base de datos, y posteriormente al efectuar consultas.

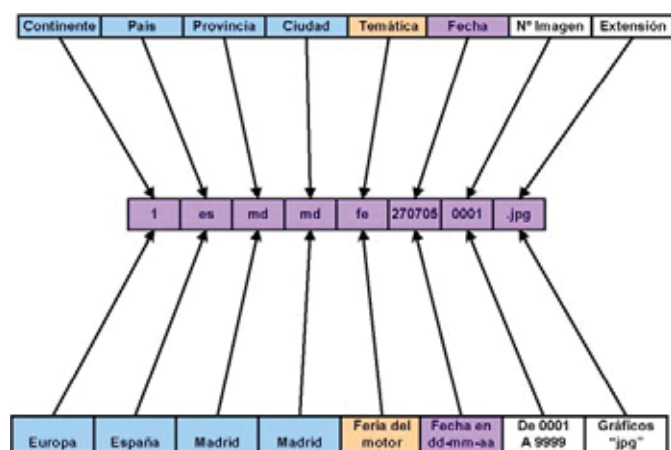


Fig. 2. Estructura de identificación de las imágenes.

C. Descripción de las imágenes o pie de foto

Al pie de cada imagen se ofrecerá una descripción o “pie de foto” de la misma, ampliando los campos de búsqueda, ya que la aplicación utiliza esta información para localizar imágenes. Se ha limitado la longitud de la misma a 256 caracteres que es el número máximo permitido por un campo *VARCHAR* en MySQL. El pie de foto estará almacenado en un fichero de texto “.txt” en el interior del directorio donde se encuentren las imágenes, el cual debe tener el mismo nombre que los ficheros que contienen las imágenes pero con la extensión “.txt”. Cuando se actualicen los directorios se agregará la información contenida en cada uno de los ficheros a su registro dentro de la tabla donde se almacena el nombre de las imágenes. Una vez introducida esa información en la base de datos ya no es necesario conservar esos archivos de texto, quedando esa decisión a juicio del administrador del sistema.

D. Sistemas de búsqueda implementados

Para facilitar la localización de imágenes se han desarrollado dos sistemas de búsqueda independientes:

- Un metabuscador por palabras (similar a Google), el cual actúa como buscador genérico de la aplicación, localizando una serie de palabras claves introducidas por el usuario desde un campo de texto en la base de datos.
- Un buscador avanzado, que permite al usuario introducir un criterio de búsqueda según unos campos preprogramados, y que posibilitarán la búsqueda geográfica, temática y cronológica. Se accederá a los mismos desde la página principal de la aplicación, tal como se muestra en la Fig. 3, como se puede observar se ha elegido un diseño muy

sobrio para la presentación de la aplicación, debido a las siguientes razones:

- Se ha buscado la sencillez y rapidez, al carecer de animaciones o presentaciones dinámicas se permite una rápida carga de la página principal, fijándose además los “frames” lateral y superior para las posteriores recargas.
- La compatibilidad entre las distintas plataformas aumenta cuanto más sencillo sea el diseño. Los colores se eligieron para mantener una coherencia con los colores institucionales de la Universidad de Cantabria, ya que será alojada en su web. El diagrama de flujo de ambos buscadores (metabuscador y buscador avanzado) implementados en la aplicación SlideFinder se muestran en las figuras 4 y 5, respectivamente.

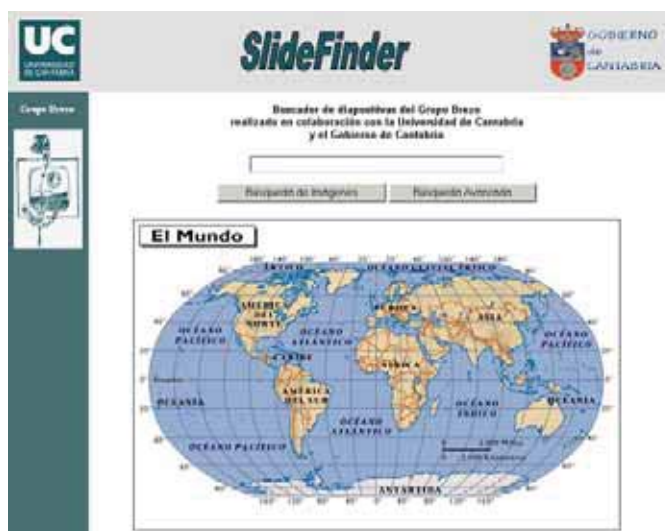


Fig. 3. Estructura de identificación de las imágenes.

Los pasos seguidos por el metabuscador por palabras o texto son los siguientes:

1. El sistema recibe la frase introducida por el cliente desde la página web.
2. La frase se procesa eliminando cualquier símbolo extraño² que pueda contener y se subdivide en palabras.
3. Comprobación si las palabras pertenecen a la tabla de excepciones, en caso afirmativo, se eliminan para reducir el tiempo de búsqueda.
4. Se buscan coincidencias en la base de datos, en el siguiente orden: geográfico y posteriormente temático
5. Las coincidencias de los casos anteriores se almacenan para realizar una búsqueda conjunta.
6. Una vez reducido el conjunto de palabras útiles para la localización de las imágenes, se procede a buscar en el campo de texto asociado como pie de página a la misma.

El algoritmo desarrollado es capaz de detectar correctamente las palabras, con o sin tildes, en mayúsculas, en minúsculas o combinación de ambas. Cualquier cambio que se desee efectuar en la forma de trabajar el buscador no conlleva excesivos esfuerzos de programación por parte de la persona que realice el mantenimiento del sistema, ya que solo deberá cambiar las directrices que se han tomado.

² Por símbolo o carácter extraño se entenderá caracteres no ASCII ó pertenecientes a páginas de códigos y alfabetos no implementadas en el ordenador donde se procesa la información (cirílico, oriental, árabe, etc.)

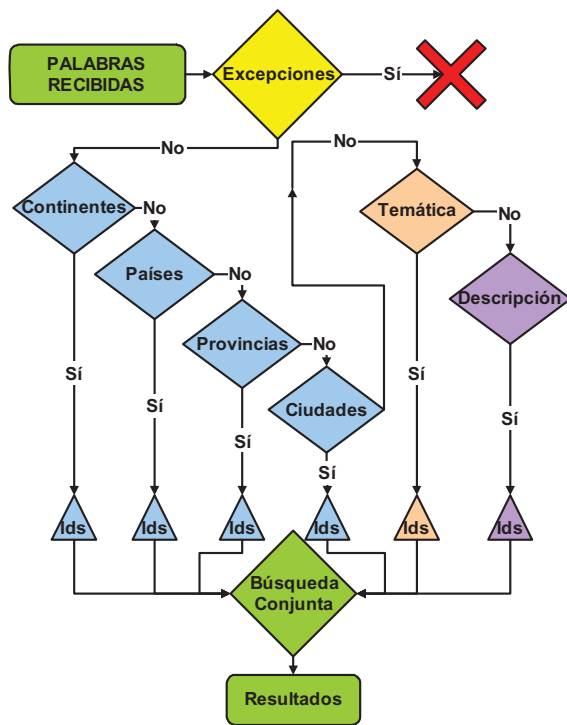


Fig. 4. Diagrama de flujo del buscador por palabras (metabuscador).

La estructura de localización del buscador avanzado se basa en los siguientes puntos:

1. Se efectúa la búsqueda geográfica, comenzando por seleccionar un continente o un país en el mapa inteligente o en el metabuscador anteriormente descrito, ambas tareas se pueden realizar independientemente, pero si se selecciona un continente, los países que mostrará la aplicación serán los pertenecientes al mismo. A partir de ahí, se puede concretar aún más la búsqueda, con la selección de una provincia y una ciudad.

2. El segundo paso consiste en la búsqueda temática, si ya existe alguna selección geográfica previa, se mostrará únicamente los temas asociados a la misma, pero sino, se podrá seleccionar cualquier temática de forma independiente.

3. Finalmente se produce la búsqueda por criterios históricos, fijado un intervalo temporal o especificando una fecha concreta. A este buscador se accede desde un enlace en la página principal, o se puede llegar a él a través del mapa de la misma página.

E. Versión “stand-alone” de SlideFinder

Dado que en algunos casos se debe distribuir parte del archivo general en formato CD, DVD, o el usuario no dispone de acceso a Internet, se ha desarrollado un instalador para una versión aligerada de la base de datos por temas. Dicho instalador solo esta disponible para Microsoft Windows, aunque ya se esta trabajando en realizarlo para otras plataformas, por ello se ha optado tanto para la versión “on line” como la “stand-alone” el mismo servidor Apache. El instalador se realiza a su vez con un lenguaje de programación NSIS [6] de código libre y permite la instalación del servidor web Apache con el motor del PHP [7-8], el servidor de MySQL, así como la aplicación SlideFinder desarrollada.

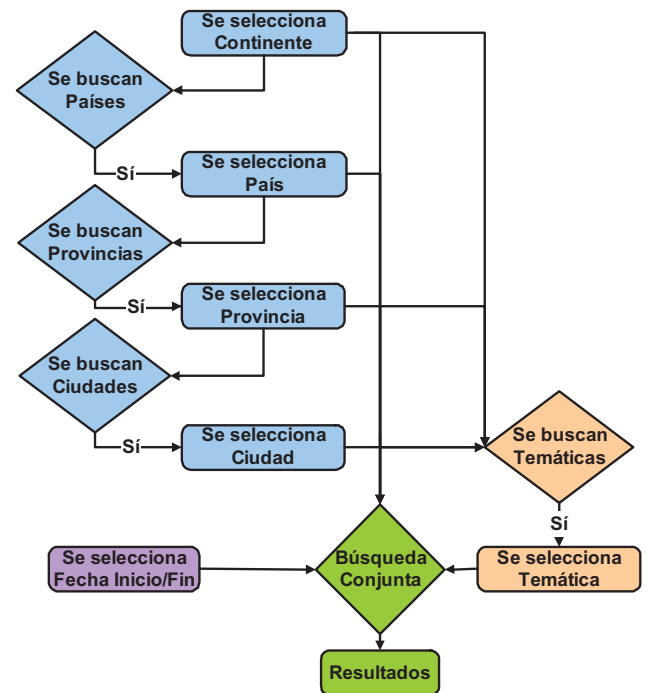


Fig. 5. Diagrama de flujo del buscador avanzado.

IV. CONCLUSIONES

La aplicación desarrollada permite la gestión y almacenamiento de imágenes, mediante el lenguaje web dinámico PHP y una base de datos MySQL, creada para poder almacenar gran cantidad de imágenes a través de una interfaz simple y rápida. La herramienta aquí presentada, ha sido desarrollada completamente mediante software libre, con lo que se pueden conseguir aplicaciones muy potentes con muy bajo coste. La idea de los autores es liberar el código fuente de la misma en cuanto finalice el convenio suscrito con la ONG Grupo Brezo. El resultado final es una aplicación que puede funcionar sobre múltiples sistemas operativos, servidores web, navegadores...

AGRADECIMIENTOS

Los autores quieren agradecer el apoyo mostrado en todo momento por los miembros de la ONG “Educación para la Solidaridad - Grupo Brezo” sin la cual el presente E-@rchivo digital no hubiese podido desarrollarse, así como el soporte económico proporcionado por la Consejería de relaciones Institucionales y Asuntos Europeos del Gobierno de Cantabria a través de la Dirección General de Asuntos Europeos y Cooperación al Desarrollo, en la persona de su director D. José Félix Calleja.

REFERENCIAS

- [1] Welling L. & Thomson L.: Desarrollo Web con PHP y MySQL., Madrid, Anaya Multime-dia (2003).
- [2] Kent A. & Powers D.: Desarrollo Web con PHP y Dreamweaver MX 2004, Anaya Multi-media (2003).
- [3] Gutiérrez Gallardo J.D.: MySQL”. Madrid, Anaya Multimedia (2004).
- [4] Página oficial de PHP: <http://www.php.net/>
- [5] Página oficial de MySQL: <http://www.mysql.com/>
- [6] Página oficial de NSIS: <http://nsis.sourceforge.net/>
- [7] Página oficial de Apache (A PaTCHy sErver): Servidor web de código abierto: <http://httpd.apache.org/>
- [8] PHP a fondo: <http://www.desarrolloweb.com/php/>